MOBILE FORENSIC TRIAGE FOR DAMAGED PHONES USING M_TRIAGE

YUSOOF MOHAMMED HASHEEM

A thesis submitted in partial

fulfillment of the requirement for the award of the

Degree of Doctor of Philosophy

Faculty of Computer Science and Information Technology

Universiti Tun Hussein Onn Malaysia

July 2016

For my beloved mother, father, wife and Daughter Radeeya

ACKNOWLEDGMENT

In the name of Allah, the Most Gracious and Most Merciful, Firstly, I would like to express my thankful to Dr. Kamaruddin Malik bin Mohamad, for his willingness to accept me as a child not just only PhD student. His guidance, support, determination, encouragement, understanding and patient along this journey are really appreciated.

I would also take this opportunity to thank Alhaji Graba Ahmed Garba for his financial support and Robert J. Walls one of the Dec0de author, for their ideas sharing, brainstorming and motivation throughout these past year.

I owe my gratitude to Universiti Tun Hussein Onn Malaysia (UTHM) for supporting this research.

I am also greatly indebted to Faculty of Computer Science and Information technology (FSKTM) and Center for Graduate Studies (CGS) of Universiti Tun Hussein Onn Malaysia (UTHM) for providing good facilities and inspiring environment for me to complete this study comfortably.

YUSOOF MOHAMMED HASHEEM, Parit Raja

ABSTRACT

Mobile forensics triage is a useful technique in a digital forensics investigation for recovering lost or purposely deleted and hidden files from digital storage. It is particularly useful, especially when solving a very sensitive crime, for example, kidnapping, in a timely manner. However, the existing mobile forensics triage tools do not consider performing a triage examination on damaged mobile phones. This research addressed the issues of performing triage examination on damaged Android mobile phones and reduction of false positive result generated by the current mobile forensics triage tools. Furthermore, the research addressed the issues of ignoring possible evidence residing in a bad block memory location. In this research a new forensics triage tool called M_Triage was introduced by extending Decode's framework to handle data retrieval challenges on damaged Android mobile phones. The tool was designed to obtain evidence quickly and accurately (i.e. valid address book, call logs, SMS, images, and, videos, etc.) on Android damaged mobile phones. The tool was developed using C#, while back end engines was done using C programming and tested using five data sets. Based on the computational time processing comparison with Dec0de, Lifter, XRY and Xaver, the result showed that there was 75% improvement over Dec0de, 36% over Lifter, 28% over XRY and finally 71% over Xaver. Again, based on the experiment done on five data sets, M_Triage was capable of carving valid address book, call logs, SMS, images and videos as compared to Dec0de, Lifter, XRY and Xaver. With the average improvement of 90% over DEC0DE, 30% over Lifter, 40% over XRY and lastly 61% over Xaver. This shows that M_Triage is a better tool to be used because it saves time, carve more relevant files and less false positive result are achieved with the tool.

ABSTRAK

Mobile forensics triage adalah satu teknik yang berguna di dalam penyiasatan forensik digital untuk mendapatkan kembali fail-fail yang telah hilang atau yang telah dibuang dengan sengaja serta yang tersembunyi di dalam storan digital. Ia amat berguna, terutamanya apabila ingin menyelesaikan satu jenayah yang sangat sensitif, sebagai contoh, penculikan, dengan menggunakan cara yang tepat dalam masa yang singkat. Walau bagaimanapun, alat forensik mudah alih triage yang sedia ada tidak berkeupayaan untuk menjalankan pemeriksaan triage ke atas telefon mudah alih yang rosak. Kajian ini dilakukan untuk menangani isu melaksanakan pemeriksaan triage pada telefon mudah alih Android yang rosak dan pengurangan keputusan false positive yang dihasilkan oleh alat forensik mudah alih triage semasa. Selain itu, kajian ini juga menangani isu bukti yang mungkin masih ada di lokasi memori blok yang rosak yang selalu diabaikan. Dalam kajian ini alat forensik triage baru iaitu M_Triage diperkenalkan dengan menambahbaik rangka kerja Decode untuk menyelesaikan masalah mendapatkan semula data pada telefon mudah alih Android yang rosak. Alat ini telah direka untuk mendapatkan bukti dengan cepat dan tepat (seperti buku alamat yang sah, log panggilan, SMS, imej, video, dan lain-lain) pada telefon mudah alih Android yang rosak. Alat ini telah dibangunkan menggunakan bahasa pengaturcaraan C#, manakala back end engine dibangun menggunakan pengaturcaraan C dan diuji menggunakan lima set data. Berdasarkan perbandingan masa pemprosesan dengan Dec0de, Lifter, XRY dan Xaver, hasilnya menunjukkan bahawa terdapat peningkatan 75% lebih dari Dec0de, 36% lebih dari Lifter, 28% lebih XRY dan akhirnya 71% lebih dari Xaver. Selain itu, berdasarkan eksperimen yang dilakukan pada lima set data, M_Triage juga mampu membuat carving buku alamat yang sah, log panggilan, SMS, imej dan video berbanding Dec0de, Lifter, XRY dan Xaver dengan peningkatan purata 90% lebih dari DEC0DE, 30% lebih dari Lifter, 40% lebih dari XRY dan akhir sekali 61% lebih dari Xaver. Ini menunjukkan bahawa M_Triage adalah alat yang lebih baik untuk digunakan kerana ia menjimatkan masa, carve fail yang lebih relevan dan pengurangan keputusan false positive dapat dicapai.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF SYMBOLS AND ABBREVIATIONS

PDA Personal Digital Assistant
ADB Android Debug Bridge
APP Application Segment
ASCII American Standard Code for Information Interchange
BHF Block Hash Filtering
CCITT Consultative Committee on International Telegraphy and Telephony
CFFTPM Cyber Forensic Field Triage Process Model
CPU Central Processing Unit
DAC Define Arithmetic Coding
DF Digital Forensics
DFRWS Digital Forensic Workshop
DHP Define Hierarchical Progression
DHT Define Huffman Coding Table
DMIF Multimedia Integration Framework
DQT Define Quantization Table
DTF Digital Triage Forensics
DVD Digital Video Disc
DVM Dalvik Virtual Machine
EC Embedded Compact
ECC Error Correction Code
EDB Exchange Database
EOI End of Image
ERC Expand Reference Component
EXIF Exchangeable Image File Format
FBUS Fast Bus
FOB Forward Operating Base
FSM Finite State Machines
FTL Flash Transaction Layer
FTYP File Type Box

GB Gigabyte
GPS Global Positioning System
IEC International Electrotechnical Commission
IMEI International Mobile Station Equipment Identity
IPD Immuno Polymorphism Database
ISO International Standard Organization
JEIDA Japan Electronics Industry Development Association
JFIF JPEG File Interchange Format
JPEG Joint Photographic Experts Group
JTAG Joint Test Action Group
MDAT Media Data Box
MF Mobile Forensics
MLC Multi-Level Cell
MOOV Movie Box
MPF Mobile Phone Forensics
MTD Memory Technology Devices
MTD Memory Technology Devices
OS Operating System
PCB Printed Circuit Board
PFSM Probabilistic Finite State Machines
RAM Random Access Memory
RIM Research in Motion
RST Restart-Interval Termination
SDK Software Development Kit
SMS Short Message Services
SOF Start of Frame
SOI Start of Image
SSH Secure Socket Shell
TCK Test Clock
TCP Transmission Control Protocol
TDI Test Data In
TDO Test Data Out
TMS Test Mode Selection
TPL Task Parallel Library
USB Universal Serial Bus
XML EXtensible Markup Language
YAFFS Yet Another Flash File System

CHAPTER 1

INTRODUCTION

1.1 Background of Study

In the last few decades, mobile phone usage was limited to only voice calls and short message services (SMS). Nowadays mobile phones, personal digital assistants (PDA), and the Internet have increasingly become a part of our daily activities due to rapid development in mobile phone technology. Mobile phone enters the personal domain and used in multimedia and personal tasks. On the flip side, these scenarios give full opportunity to innovate ways of perpetrating criminal offenses (Abdullah *et al.*, 2014). To compound the problem further, crimes committed using mobile phones are laborious to prove especially "damaged" mobile phones. Digital forensics (DF) is a platform for recovery and investigation of data found in digital evidence, mostly in helping investigations related to mobile phone crimes (Abdullah *et al.*, 2014). Also, DF has to preserve, identify, press out, record, render and analyze evidence contained in digital storage (Ahmed *et al.*, 2013), (Chang *et al.*, 2013).

DF began in a limited role however as the years go by coupled with the boom in mobile phone technology and popularity, DF experienced its so-called Golden Age from 1999 to2007. Digital forensics played a significant part in solving mobile phone crime cases against such as, drug dealing, child trafficking, and illegal arms trade. Mobile phone capabilities increased in public presentation, memory capability and multimedia functionality turning phones into data pools able to support a wide range of personal information (Curran *et al.*, 2010).

DF is significant to reconstruct evidence left by criminals. Furthermore, DF has a given domain called Mobile Phone Forensics (MPF) for investigation and the extraction of data or information from a mobile phone device. It is defined as the scientific recovery of digital evidence from mobile phone using proven and accepted

methods.

According to Owen & Thomas (2011) MPF is one of the hardest and the most challenging fields of DF. Data held on mobile devices can be useful and important to law enforcement agencies when carrying an investigation in either civil or criminal transactions. There are two ways of recovering digital evidence, using, traditional data recovery, and, file carving. The traditional data recovery is a usual technique applied to retrieve digital information where the metadata or file allocation table exists. Meanwhile, file carving was introduced in criminal cases where traditional data recovery techniques are unable to provide assistance. However carving is used to identify the operation of extracting raw image or dump file from digital devices. One of the most important features which carving techniques have over traditional recovery is that it allows to analyze a block or a set of blocks against characteristics of a specific file format and/or its contents (Muhammad & Ashraf, 2012). Another advantage which carving techniques hold over traditional recovery is that various acquisition methods are available for extracting evidence from mobile devices, such as, Manual, Logical and Physical methods (Grispos *et al.*, 2012).

Each one of the acquisition method use different features of the device for pressing out the selected quantity of data. Manual extraction is identified, as anything an individual is capable of acquiring by interacting with the gimmick itself. This procedure may consist of two separate phases where one would keep a log of the actions taken while interacting with installed applications to simulate the existing information. Secondly, cameras can be employed in order to prove the state of the device. Logical extraction retrieves a bitwise copy of entities such as files and directories that reside within a logical storage and it "provides context information for objects such as, date-time stamps and location within the file system of the target mobile device" (Casey & Turnbull, 2011).

This extraction method mainly concerns data that has not been edited and is reached by accessing the file system of the device (Barmpatsalou *et al.*, 2013a). According to Kalva *et al.* (2013) and Breeuwsma & Jongh (2007) "True physical extraction can either mean physically removing memory from the gimmick, using hardware techniques like Joint Test Action Group (JTAG), in order to extract data from the device, or use an adapted bootloader to reach a low-level access to the device". These techniques "are not only technically challenging and involve partial-to-full disassembly of the device, but they require significant post-extraction analysis to reassemble the file system" (Hong *et al.*, 2013).

Android is used in this research as the platform has been applied across a wide range of devices, predominantly mobile phones, thus bringing together a unique common software feature to the diverse set of devices independent of carrier and manufacturer (Vidas *et al.*, 2011). The Android platform is already the most current among

mobile communication devices. There are two major forms of memory present in Android devices, namely, volatile RAM and non-volatile NAND flash storage. RAM is used to load and operate the critical parts of the operating system (OS), applications, and information. RAM being volatile does not preserve its data once the phone is damaged or powered down. However, NAND flash memory is non-volatile and data is saved even if the de- vice is powered down. Android delivers an exceptional method to manage application memory (Thakur, 2013).Nevertheless, the forensic extraction tools available to most forensic analysts do not provide direct access to dead mobile phones on the Android platform and are limited to acquiring data in the bad block, and erased block through a hardware concept layer(Casey *et al.*, 2011). Akkaladevi *et al.* (2011) mentioned that forensics toolkit's lack of performance speed during the investigation procedure. The traditional approach utilized a single workstation to perform digital investigations against a single source medium, which is time-consuming .

Furthermore, a previous research used the 4-step mobile forensic workflow based on device identification, acquisition, analysis, and reporting, was found to be inadequate to facilitate current investigations. An intermediate step was introduced, which is mobile forensics triage, positioned between acquisition and examination, with the goal of finding the file of interest and reduce the number of irrelevant files. Mobile forensics triage is a partial forensic examination conducted under significant time and resource constraints. The results from a triage analysis are used to assist the forensic investigator in determining whether the digital media may hold any evidence of value. The mobile forensics triage results assist a forensic investigator in prioritizing media in difficult investigation circumstances.

1.2 Problem statement

As mentioned earlier, one of the major problems in the field of MPF is that forensic toolkits lack performance speed during the investigation procedure. Nevertheless, the traditional approach utilise a single workstation to perform digital investigation against a single source medium, which is time-consuming. This leads to a significant need for triage examination in the DF field in other to speed up the work in the laboratory and on site. Triage examination enables the investigator to perform an efficient and accurate analysis on any device. There remains a minimal availability of MF triage tools today, and presently some issues still linger surrounding the tools, one of it is known as "DEC0DE," where the software failed to perform triage analysis on damaged mobile phones. Secondly "DEC0DE", returned over 6.2 million results during MF triage extraction, of which only about 12 thousand were relevant, Walls & Levine (2014) observed this produced a high number of false positives by the said tool.

Third, "DEC0DE" only focused on feature phones, for instance, phones with less capability than smartphones, moreover, the maximum memory capacity "DEC0DE" can handle is 48MB, which according to Walls *et al.* (2011) which is not good enough.

"DEC0DE" failed to fix the challenges of extracting the physical memory dump from phones, which is an input needed in data extraction, since the issue of bad block in NAND flash memory is not addressed and so there is no 100% integrity of any data analyzed by "DEC0DE", due to lack of any proof if the dump file had not been altered or fully extracted before given to "DEC0DE" for triage examination. Lastly "DEC0DE" had not managed to extract multimedia files. In this thesis, a novel MF triage tool called M_Triage were introduced to overcome the above problems which the research will extend the tool to in order to be able to handle damaged mobile phones. However, data set from the Digital Forensic Research Conference, 2010 (DFRWS 2010) will be used to validate this research.

1.3 Aim and objectives of the study

The aim of this research is to propose a technique during triage extraction, which will handle damaged mobile phones, and also manage a scenario where there is a bad block that hides evidence within the NAND flash memory of the damaged Android-based mobile phones.

The objectives of this research are:

i To propose a new forensic triage technique for handling damaged Android-based mobile phones.

ii To implement the proposed technique.

iii To compare the proposed triage techniques based on the average time taken to successfully retrieve valid address books, call logs, SMS messages, images, and videos.

1.4 Scope of the study

This research focuses on "Dead-forensic" of damaged Android-based mobile phone and centers on the triage extraction of the address - book, call logs, SMS messages, images and videos, including the meta-information. This research excludes handling of fragmented files and generation of full in-depth analysis result based on evidence examined with M_Triage. The proposed approach concentrates on triage techniques

based on the average time taken to successfully carve valid address books, call logs, SMS messages, images, and videos on damaged "Samsung" products only.

1.5 Organization of thesis

The following chapter provides a background information on data retrieval and file carving. It describes different techniques of file carving and compares with one another existing tools that handle physical extraction. Chapter 3 discusses M_Triage framework and algorithms, data-set preparations and handing of NAND flash memory with a bad block. In addition, in chapter 3 there will be a discussion on how to perform the triage extraction for the address book, call logs, SMS messages, images and videos on damaged android-based mobile phones. Chapter 4 discusses the important processes in implementing M_Triage for damaged android-based mobile phones. Chapter 5 talks about the experimental process on the selected models of some damaged mobile phones. Chapter 6 will discuss the result of the triage examination processes and compare the implemented techniques with other triage tools. Lastly, Chapter 7 concludes the research and provides suggestions for future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Early research in this area has centered on learning techniques and overall forensic analyses of smart devices (Thing *et al.*, 2010a). Al Barghouthy *et al.* (2013) stated that recent scientific inquiry experiences focused on distinct types of smartphones, investigating the methods that could be employed to acquire and analyse the internal memory of the gimmick and the information that could be pulled from each device. According to Jansen & Ayers (2007)), MF is the science of recovering digital evidence from a mobile device under forensically sound conditions using accepted methods. Forensic investigators commonly start with telephone numbers dialed, answered, received or missed stored phone numbers of people whom the mobile phone user may know, and text messages sent, received or deleted. Mobile phones are very important to people who utilize such knowledge to convey with others and to organize day-to-day activities. Hence, it can be a very significant source of data for legal prosecution and corporate agencies during criminal investigations. These mobile phones hold vital and significant information that can be utilized as proof in the court of law. Granting to the National Institute of Justice in the USA (2010, p. 4), "digital evidence is directly employed to prosecute all different sorts of crimes" (Vasa, 2013). This type of evidence is satisfactory in a courtroom of law as affirmed by the ISO/IEC International Standard (ISO/IEC 27037:2012), which will certify the reliability and credibility of such evidence during court cases and legal arguments (Meyers & Rogers, 2004).

Offenders can use mobile phones to organize and achieve wrongdoings such as homicide, burglary, drug dealing, money laundering, fraud, identity stealing, hacking, pedophilia, child abuse, sexual harassment and including electronic crimes (Gottschalk, 2010). Even so, many tools are available for mobile phones to perform forensic analy-

sis, but these tools are not always compatible with all the manufacturers and different types of mobile phones (Brinson *et al.*, 2006). However, the development of new criminal techniques in "dead" mobile forensic presents some drawbacks for law enforcement in the domain of digital mobile forensic. As a consequence, crime scene investigators cannot always apply dead digital forensics successfully to collect sufficient evidence to lead to a conviction (Casey *et al.*, 2011). In the year 2011, Lessard *et al.* (2011) mentioned that unitary of the major difficulties in the area of mobile forensic is the universal lack of hardware, software, and/or interface standardization within the industry. This fact makes forensic processing a heavy job, especially for integrated research. Mobile phone forensics is a challenging field due to the quick changes in engineering science. Various types of mobile phones exist in the world today .and manufacturers lack standardized methods of storing information. Most mobile phones use closed operating systems and has proprietary interfaces. To overcome this challenge there is always a need for the development of new forensics tools and techniques (Barmpatsalou *et al.*, 2013b).

2.2 Digital forensics

DF has a dedicated field for investigation and the extraction of data or information from a mobile phone device called MPF. It is defined as the scientific recovery of digital evidence from mobile phones using proven and accepted methods (Ayers *et al.*, 2014). DF plays an important role not only in helping in solving cases against mobile phone crimes like drug dealing, child trafficking, and arms deal. Mobile phone capabilities increase in performance, storage capacity and multimedia functionality turning phones into data pools that can hold a broad range of personal information (Curran *et al.*, 2010). DF thus becomes an important tool to rebuild the evidence left by criminals.

According to Owen & Thomas (2011)), MPF is one of the toughest and the most challenging fields of DF. And from an investigative perspective, digital evidence recovered from a mobile phone can provide a wealth of information about the user, and each technical advancement in capabilities offers greater opportunity for recovery of additional information (Thing *et al.*, 2010b). However, the software applications for mobile forensics available today are not 100% forensically sound. The reason is that they use command and response protocols that provide indirect access to memory (Lessard *et al.*, 2011).

The golden age of DF occurred from 1999 to 2007. This is when DF emerged in the efforts to reduce the rate of cyber-crimes. It is being used as the tool to look into the past through the recovery of remaining data that was thought to have been deleted through the recovery of email and instant message. The two important fields in DF

are data recovery and file carving which is explained as file recovery techniques that make use of the file system information that remains after deletion of a file. Using this information enables many files to be recovered. For this technique to work well, the file system information needs to be correct. If the file system is incorrect, the files cannot be recovered. If a system is formatted, the file recovery techniques will not work either. File carving deals with raw data on the media and does not use the file system structure during its process. Carving makes use of the internal structure of a file. A file is a block of stored information like an image in a JPEG file.

2.3 Digital forensics branches

Digital forensics can be divided into sub-branches based on the type of the investigated device, which is, environment, media and digital artifacts. These sub-branches mainly are computer forensics, network forensics, mobile device forensics and database forensics as shown in Figure 2.1(Karabiyik, 2015). In the following subsections, the research will dis-cuss mobile forensics techniques in detail and explain other digital forensics branches briefly. Nevertheless, the main focus of this research is on forensics triage extraction for handling damaged Android-based mobile phones.

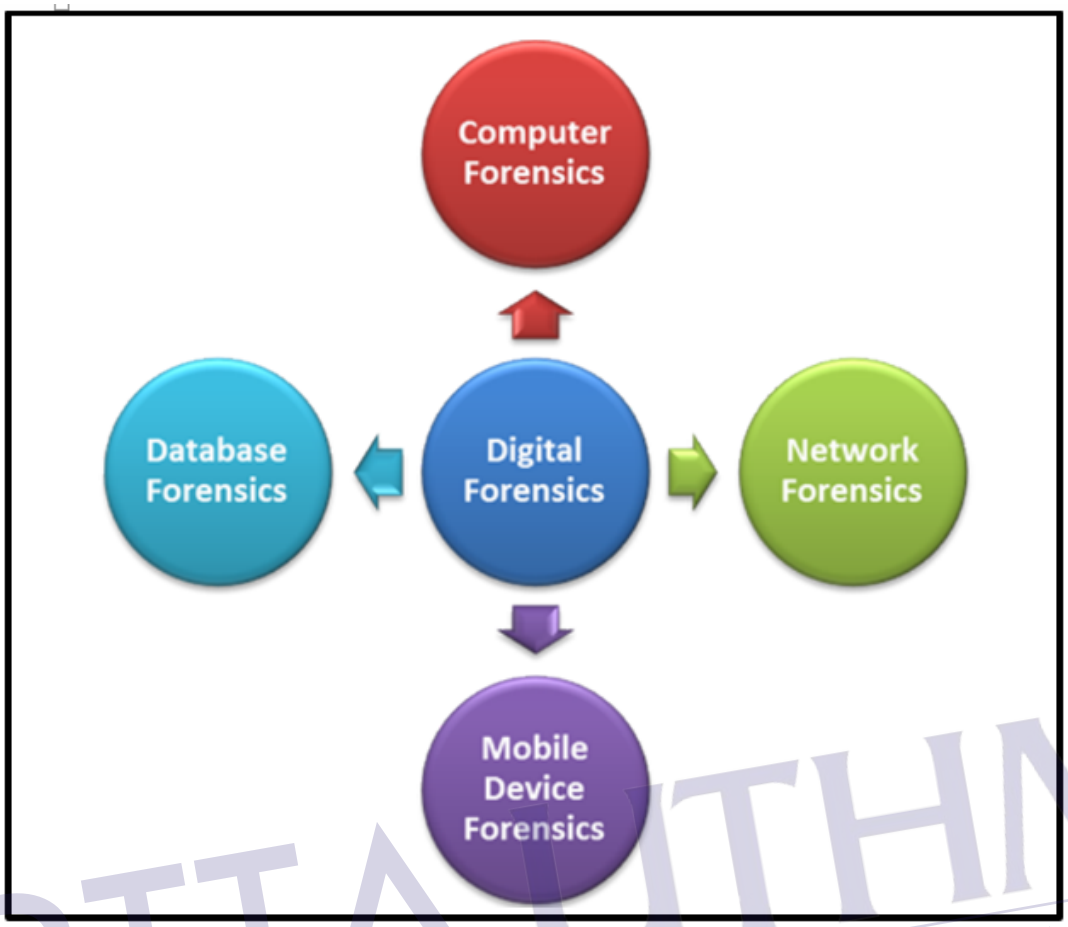

Figure 2.1: Digital forensics branches Karabiyik (2015).

2.3.1 Mobile forensics

With the increased emphasis on social security issue, crime is significant when it comes to the utilization of smart phone technologies, DF provide the technical skills to collect evidence for the court to review and judge cases. Digital tools has changed daily, and digital devices are used pervasively such as computers, mobile phones, digital cameras, hardware, storage devices. Presently, DF is widely used in the area of network forensics, mobile forensics, computer forensics, and memory forensics (Ahmed *et al.*, 2013). According to Ayers *et al.* (2014), the science of extracting digital evidence from a mobile phone under forensically sound conditions using accepted methods is known as MF. By default, the field of MF is challenging due to the fact that smartphones have limited processing and memory resources, different central processing unit (CPU) architecture and a variety of well-secured OS versions compared to those of a personal computer, making forensic processing a difficult task. Mobile phones can hold personal information including call history, text messages, e-mails, digital photographs,

videos, calendar items, memos, address books, passwords, and credit card numbers. These devices can be used to communicate, exchange photographs, connect to social networks, blogs, take notes, record and consume video and audio, sketch, access the Internet, and much more (Dubey, 2013). Smartphones have become an integral part of people's daily lives, and as such, they are prone to facilitating criminal activity or otherwise being involved when crimes occur. The importance of mobile phone from a forensic viewpoint is that they hold deleted information even after an individual has attempted to render it unrecoverable. The underlying reason for this persistence of deleted data on mobile phone is in the use of Flash memory chips to store data.

Yang & Yen (2010) put emphasis on live and dead forensic analysis that can be carried out by saving the required scripts and different tools like Autopsy, FDumper, and Scalpel, Fundl, etc on a USB or DVD. Such method can help in performing live analysis of a running compromised system by plugging in the DVD/USB into the system. The script/tools stored on the DVD/USB when launched will collect the volatile information such as opened ports, user login history and active services etc. from the memory of system and store it on the USB (Bashir & Khan, 2013). Hence, MF is described in two categories, which are "Live" and "Dead" mobile forensics.

2.3.1.1 Live mobile forensics

Live mobile forensics, sometime referred to as live incident response on digital devices, is a technique to extract memory, system processes, on powered devices such as mobile phone, in addition, live mobile forensics plays a vital role during forensics examinations due to the potential availability of digital evidence in the volatile memory such as running processes. Live mobile forensic investigation mainly targets the volatile data which can only be extracted from a running OS; hence, the term "live" is created for such type of examinations or else the information cannot be extracted from a "dead" OS whose power is down. Conducting live mobile forensics has become compulsory in the modern era (Bashir & Khan, 2013). Mrdovic *et al.* (2009) proposed and performed live mobile forensics analysis concurrently, which enhances the understanding of events and provides additional understanding into the current state of mobile phones for examination.

Live mobile forensics investigation is done through visualization. A research by Thing *et al.* (2010b) where he investigated the dynamic behaviour of the mobile phone's volatile memory and presented an automated system to perform a live memory forensic analysis for mobile phones (Azadegan *et al.*, 2012). Furthermore, Thing mentioned that live memory forensics has an even more important role to play. As mobile phones are becoming gradually prevalent and are always growing into "smarter"

devices (i.e. smartphones with higher processing power and enhanced features), consequently, the abilities to perform in-depth forensics on these devices are also equally vital.

2.3.1.2 Dead mobile forensics

Al-zarouni in his research mentioned that digital investigations can involve dead and/or live analysis techniques in MPF (Al-zarouni, 2006).In dead mobile forensic analysis, the target device is powered off, whereas non-damaged or damaged is where the mobile phone cannot be powered on in order to perform DF examination. In addition Crisalis, 2013 stated that the approach which data are extracted from a powered down systems is known as dead forensics. Furthermore, Cantrell *et al.* (2012) mentioned that "Dead" examination is shown on evidence that has previously remained powered off either because the mobile housing it has been booted into a digital triage environment or it has been seized and powered down for a proper examination. Table 2.1 provides the description of powered off/ non-damaged and damaged mobile phones (Fp-sec crisalis, 2013).

Table 2.1: Description of powered off, non-damaged and damaged mobile phone.

Powered off, non damaged mobile phone	**Damaged mobile phone**
In “dead” MF, phones which are not powered on, due to low battery during forensics analysis are con ceder as non-damaged mobile phone.	mobile phone which are not able to be powered on, due to water damage, fire damage, or some technical error by the printed circuit board (PCB) during forensics analysis are con ceder as “damaged” mobile phone.
Mobile phones which must be powered off before conducting forensics analysis required by the tool (i.e. UFED physical analyzer), are con-ceder as non-damaged mobile phone.	mobile phone which can be powered on, but the screen is damaged to a point where the examiner cannot view the content of the mobile phone during forensics examination are con-ceder as “damaged” mobile phone.
mobile phone which are pass-worded by the criminals, and required additional software to by-pass the password before and during forensics examination are con-ceder as “non-damaged” mobile phone.	mobile phone which the USB ports is bad due to water or fire damage, and data on the mobile phone cannot be access through blue-tooth or wireless connections during forensics examination are con-ceder as “damaged” mobile phone.

2.3.1.3 Other digital forensics branches

Before embarking on the explanations of the other DF branches in addition to mobile phones, it is significant to note that, in DF there are different technologies applied in our

routine lives and these technologies do not belong under a single branch. For

instance, cloud forensics may also be deliberated under another digital forensics branch at the convergence of computer and network forensics since both environments are applied during its use. Therefore, the research will be discussing only the main digital forensics branches in this segment.

2.3.1.4 Network forensics

Network forensics is an offshoot of digital forensics, which concentrates on studying computer networks and their dealings in order to find anomalies and malicious operations performed on the network. The primary aim is to accumulate data and possible evidence related to the investigation in order to give it to the tribunal of jurisprudence (Karabiyik, 2015). Nonetheless, network forensics is in some way different than the other digital forensics branches due to the type of information that is dispensed with. Most of the information in the network is dynamic and explosive which makes the investigation quite hard to conduct because when the web traffic is transmitted the data could be missed (Casey, 2004).

2.3.1.5 Database forensics

Database forensics deals with forensic analysis of databases and sensitive data in them as well as the metadata that describes the data itself (Olivier, 2009). Database forensics needs a close attention because of the quantity of information required to be analysed for such crimes that may necessitate databases to be practiced upon. In the case of financial crime, an investigator may require to analyse a tremendous amount data in companies' databases.

2.3.1.6 Computer forensics

Disk forensics is a sub-category of computer forensics, and it specifically targets hard disk drives as the origin of the probe to extract forensic information. During digital forensic investigation, investigators will look at different components of the computers that are affected in the case, where digital evidence resides or can be deleted or obscured (Kizza, 2010).

2.4 Crimes involving mobile phones

Blokhuis & Puppe (2010) outlined a crime involving a mobile phone, where, "after an extensive undercover operation, a known arms dealer named Monsieur Victor, commonly identified as "The General", was lured out of hiding and apprehended in the Netherlands. He had anticipated a meeting to settle a big sale of weapons, including armored combat vehicles, missiles, attack choppers, and assault rifles. Instead, he met with the police. When he realized the operation, he threw away a mobile device in a nearby canal. The device was subsequently retrieved by scuba divers, and was found to be a Sony Ericsson K800i Cybershot". "Ronald Williams killed his wife Mariama, apparently in a rage after finding out that she had an affair. Unbeknownst to Williams, his mobile phone pocket-dialed his wife's mobile phone during the crime and the call went to voicemail. The recording of his wife's voicemail captured him saying that he was going to kill her, pursued by her screams and their 2-year-old daughter pleading with Williams to stop" (Casey & Turnbull, 2011).

"Investigation into the death of 15-month-old Charlie Hunt revealed that his mother's boyfriend, Darren Newton, had beaten him over several months (Williams,2010). Incriminating evidence was found in the shape of videos that Newton had taken using his mobile device of himself assaulting the child. The videos, apparently taken over a period of months, showed Newton repeatedly slapping the child on the head for prolonged periods. On November 19, the final time that Newton assaulted Charlie Hunt, the child passed away, and Newton was sentenced to life in prison for murder" (Casey & Turnbull, 2011).

"Although drug dealers were using cheap, disposable mobile devices to operate their criminal enterprise, digital investigators were able to extract information from these devices to capture over 20 drug dealers in Medford, Oregon. In addition to linking drug dealers based on call history recovered from mobile devices, digital investigators recovered photographs of individuals handling or selling drugs" (Casey & Turnbull, 2011).

2.5 Data integrity

In any forensic investigation, data integrity means that correctness of data starting from evidence collection to reporting in the court of law (Christi *et al.*, 2011). Data integrity check proves correctness of data, which means there is no alteration to the evidence at any point of situation in forensic investigation. Data integrity can be achieved in different ways for example hashing method which can be applied on any digital forensics tool so that it can provide data integrity check (Christi *et al.*, 2011).

The current popular hashing algorithms available are MD5, SHA-1 and SHA-256.Walls *et al.* (2011) utilized SHA-1 in his block hash filtering algorithm, the author hashed every block that contain relevant data in order to preserve the integrity of the extracted data. Another author, Law, 2011 presented a cryptography model to protect data secrecy during digital investigation. In this model, investigators examined the bit stream image instead of examining the complete memory contents on storage media. Then encryption key is generated by the information owner (Law *et al.*, 2011). Naresh (2013) proposed a novel and effective means of storing SMS messages in Android mobile devices in the event if any SMS message is brought as an evidence in the court of law.

2.6 Operating systems

A factor of heterogeneity which is an impediment against the development of a common MF framework is the existence of different OSs (mobile platforms). Current market share gives Android and iOS the dominant percentages (Becker *et al.*, 2012). Other OSs, such as Blackberry and Windows also remain as a popular option. As informed earlier, the research will discuss in detail about Android operating system and the related literature. The other OSs will be briefly explained in the study.

2.6.1 Android OS

Android is an open source mobile device OS developed by Google, based on the Linux 2.6 kernel. Due to its proven driver model the Linux kernel was chosen, existing drivers, computer memory and procedure management, and networking support along with other core OS services. It has also made its own Java Runtime engine, optimized for the limited resources available alongside a mobile platform, called the “Dalvik Virtual Machine” (DVM). Finally, the application framework was created in order to provide the system libraries in a concise manner to the end-user applications (Quick & Alzaabi, 2011a). However, the basic Android architecture is Linux Kernel, and it is composed of five primary elements (Faheem *et al.*, 2014). These are presented and explained in Figure 2.2

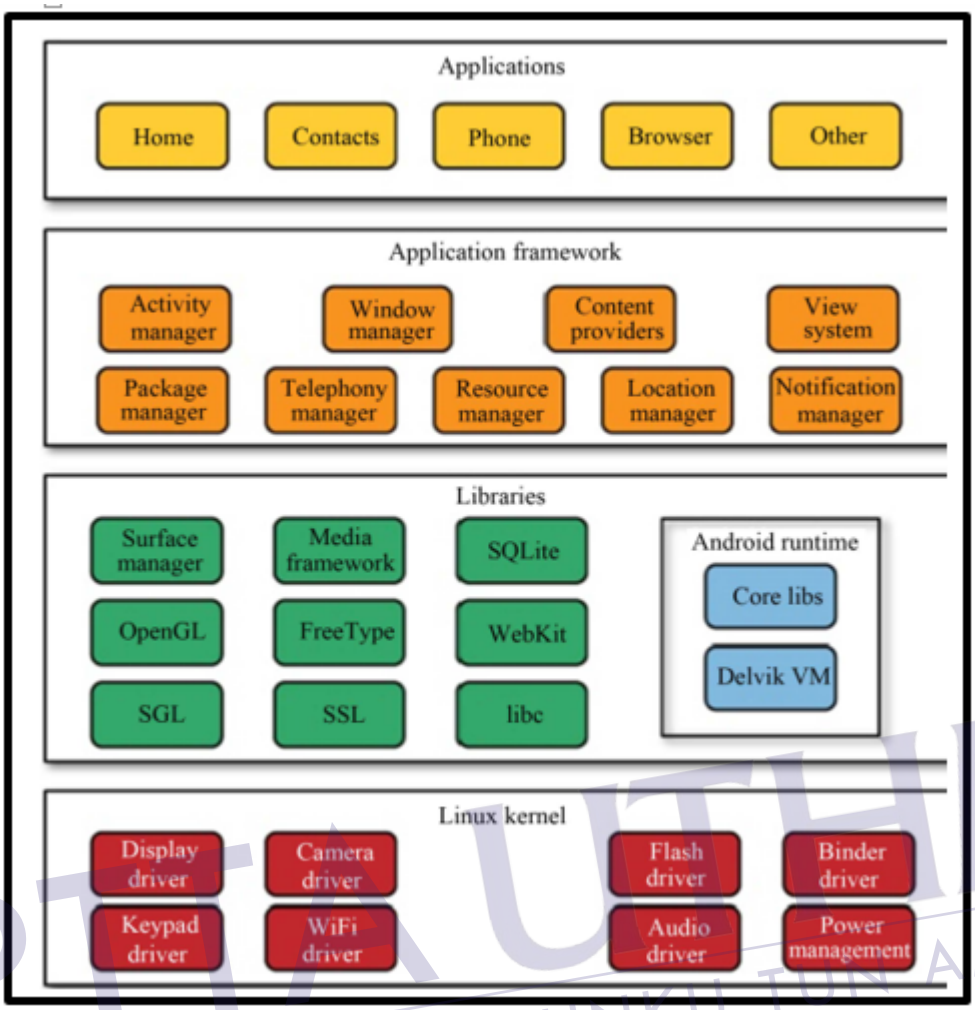

Figure 2.2: Android OS architecture Faheem *et al.* (2014).

In Barmpatsalou *et al.* (2013b) mentioned that the next tier of the Android architecture is the sphere of the libraries, split into applications and Android runtime ones. The former category provides the appropriate infrastructure for applications to move decently, such as binaries and graphics support, while the latter consists of the DVM and the core libraries that provide the available functionality for the applications (Yates *et al.*, 2010). Its primary aim is the foundation of a stable and secure environment for applications execution. Each application goes in its own sandbox (virtual machine). Thus, it is not affected by other applications or system functions. Using certain resources is entirely permitted by special privileges. This path, a satisfying layer of protection is maintained. While the Android Runtime Libraries are written in Java (Yates *et al.*, 2010)), DVM translates Java to a speech that the OS can perceive (Simao & Sicoli, 2011). The respite of the architecture consists of the Applications Framework and the Applications Layer that manage the general application structure, such as containers, alerts and the applications themselves. Referable to the small flake size, non-volatile nature and energy efficiency, NAND flash memory was selected to equip

Android devices for storage purposes (Hoog, 2011) (Zimmermann *et al.*, 2012).NAND flash memory needed a file sys- tem that was "aware of the generic flash limitations and adopt these into account along the software level when reading and writing data from and to the chip"(Zimmermann *et al.*, 2012). Yet Another Flash File System 2 (YAFFS2) was the first file system implemented for devices running Android. As years go by, many issues emerged concerning system performance, velocity of input/output activities and large files coverage. As mobile device architecture tends to stick to the path of desktop information processing systems and acquire multiple core processors, there has since been another obstacle (Kim *et al.*, 2012).

Right before the release of their version 2.3 of the OS (Gingerbread), the file system was replaced to EXT4. The specific file system, apart from successfully coping with the feeble points of YAFFS2, is enhanced with the journaling event function,Kim *et al.* (2012), which provides recovery options and facilitates acquisition of unallocated files. Android provides potential developers with the SDK (Software Development Kit), which includes a very significant tool for forensic and generic purposes, the Android Debug Bridge (adb). Adb uses a TCP or USB connection between a mobile device and a data processor. The appropriate software is installed on both sides in order to acquire debugging information, start a shell session with the provided interface, initiate file transactions, and add or transfer applications (Hoog, 2011) (Simao & Sicoli, 2011) (Vidas *et al.*, 2011).

Since adb grants a terminal interface, actions like rooting and memory image extraction can be easily done. NAND flash memory was incompatible with the Linux-based core. A new technique had to be implemented to provide the software components with the ability to access the flash memory areas (Vidas *et al.*, 2011). The Memory Technology Devices (MTD) system was one of the facilities serving as an intermediary between the pith and the file arrangement and is present in many Android devices. Handsets that do not support the MTD system usually utilize the plain Flash Transaction Layer (FTL) that enables communication between the two parts Hoog (2011). Although there are no limitations concerning the MTD numbers or types, a certain measure had been swept up from many device manufacturers (Lessard *et al.*, 2011)(Hoog, 2011) (Vidas *et al.*, 2011).

Recently, Lai *et al.* (2011) carried out a live forensic acquisition procedure, established on commercial forensic suites through cloud computing, designed for Android devices. After a brief introduction to the Android OS and forensic legislative guidelines, Jansen & Ayers (2007), they enumerated the prevailing characteristics of cloud computing and how it could keep a strong setting, suitable for conducting forensic acquisition. They reasoned that a cloud computing service, Google Cloud service in their shell, could fulfill a variety of weather, such as security prerequisites, browser-based applications, bigger storage capacity and lack of time and location limitations.

Another experimental research conducted by Quick & Alzaabi (2011b) used logical and physical acquisition techniques and tools (adb pull, NANDDump, xRecovery, and yaffs2utils) on a rooted Sony Xperia 10i device. Logical acquisition was not able to find the full size of the file system, while physical, as expected, achieved a bitwise acquisition of the flash memory. Physical acquisition with sparse data included follows a different attack, since the researchers needed to rebuild the YAFFS folder structure.

Simao & Sicoli (2011) offered a forensic acquisition framework for the Android OS. Their framework were shown in flowchart form, since there had been many different states of target devices, such as rooted or not, switched on or off, upon access control or otherwise. Even if their model can be applied to many scenarios, it is lacking some important elements concerning a routine investigation. The extra information in their proposal concerned acquisition of damaged devices and fragmented memory page analysis.

In parliamentary law to corroborate the strength of the model, Simao & Sicoli (2011) conducted experiments on devices with different conditions and figured out that the proposed system was used. Nevertheless, they acknowledged that further research should be conducted so that the framework can be kept up-to-date with the future versions of Android. A more enhanced version of the existing model was presented by Park *et al.* (2012), even though their end was not the implementation of a framework.

Sylve *et al.* (2012) referred to a lack of subject areas applicable to the physical acquisition in the context of MF. They played up the importance of this issue, unlike most other research which bypassed the issue. The researchers presented "a methodology for acquiring complete memory captured from Android, code to analyze kernel data structures and scripts that allowed analysis of a number of user and file-system based activities". Besides, they enumerated the existing methodologies on volatile memory analysis for Linux and Android OSs and compared the capabilities of the corresponding tools. Before continuing with acquisition, they had to face the rooting challenge. They thought it was a necessary evil because the code, which was expected to render the memory image, had to access the device core. Location had also been an attempt for memory acquisition by the purpose of methods intended for the Linux OS. The consequences of their experiments showed that Linux oriented techniques were incompatible with the Android OS, since a plentiful of bugs appeared, such as non-existing functions, limited size of offsets supported by the (well-known) *DD* command as well as insufficient percentage of acquired memory.

Vidas, 2011 took research to a different point, facing the challenge of a forensic acquisition of devices protected by a screen lock. Since a brute force attack on the device was not a preferable method and may contribute to further blockage and inevitable data modification, another technique had to be carried out.

In this direction, booting with a recovery image could easily bypass any form of active lock code. After enumerating the criteria for a proper forensic analysis, they proposed an acquisition method based on the usage of an acquired recovery image and *adb* software on the workstation the device is plugged in to. Single of the MTD file present in the base folder of Android devices, known as *mtd*3 (recovery mode boot) was important in the acquisition process of the recovery image (Vidas *et al.*, 2011).

2.6.2 Blackberry OS

Blackberry OS devices are developed by Research in Motion (RIM) company and have a diverse range of popularity worldwide. Few things concerning the Blackberry OS itself and its constituents are known from official sources, since the producer does not furnish sufficient support. Nevertheless, significant amounts of information concerning support were obtained via reverse engineering. These acquisitions are certain to spark further inquiry (Barmpatsalou *et al.*, 2013b).

The social system of the OS itself made the immuno polymorphism database (IPD) file as the first berth for a potential researcher to look for important information. An early effort to acquire the contents of the IPD file in terms of backup retrieval was carried out by developing IpdDump, a Java application first released in 2008.

It was initially capable of solely extracting SMS messages,Fairbanks *et al.* (2009),but newer versions are claimed to confirm attainment of other characters of user data as well. The final stable version was released in 2009, while a release candidate was made available in 2011. Its function was summarized to parsing, “extracting and exporting all types of records into customized open text formats as well to cut records like service books and contacts” (Barmpatsalou *et al.*, 2013b).

2.6.3 Windows Mobile OS

The Windows Mobile OS is the evolution of Windows embedded compact (CE), used primarily on handheld devices, such as palmtops and PDAs (Kumar *et al.*, 2012). The Windows Phone OS is its replacement, with many structural components of forensic importance in common, such as exchange database (EDB) files (Kaart *et al.*, 2013). It is a Windows-based system, with similar properties specially modified so as to apply to the nature of nomadic devices. One of the basic examples in this category is its file organization.

Klaver (2010)work, has been an influence to many future researchers since not only it introduced revolutionary techniques in the MF field, but also talked about

the most significant parts of the hardware and software related to them. His work concerned the issue of physical acquisition mechanisms on smartphones incorporating the Windows Mobile OS, ver. 6.0. The most significant attribute of forensic importance were the constructors and the RAM heap present in all the Windows Mobile devices. The central use of boulders is system booting. In the forensic ecosystem, they are also used for making away a physical binary image of the memory of the device with effective or fruitless outcome

2.6.4 iOS

iOS was first released in 2007. It is a UNIX-based OS, part following the architecture of the Mac OS X equivalent. The principal storage device of a mobile phone running the iOS is divided into two divisions. The beginning holds the OS fundamental structure and the applications, while the second comprises all the user-manipulated data (Husain *et al.*, 2011).

Zdziarski (2008) reached the breakthrough of implementing a physical acquisition technique, particularly designed for the iOS. There are no other similar efforts in literature at least for the time being. It was generally claimed that even the jailbreak technique he used was superior to other widespread ones (Hoog & Gaffaney, 2009). Specifically, the unique characteristic of the method focused on changing an amount of information in the system partition but left the user data partition untouched.

In whatever instance, the ideal state of no data modified had not been reached; a forensically sound image of the user data though had been a breakthrough. And so, he booted the test device with a recovery toolkit Zdziarski (2008),booted the test device with a recovery toolkit , which held the essential software enabling him to obtain a bitwise copy of the memory image. Some other famous feature was the usage of secure socket shell (SSH) in the recovery toolkit for making an encrypted bridge between the twist and the workstation. Going around the security code was accomplished by the induction of the iPhone Utility Software while other recovery/viewing programs were employed to con- vert the acquired image to a human interpretable format. Zdziarski contributed a major improvement in the iOS forensics field. The research needs to be extended, since new versions of the OS are implemented and previous techniques may have been already outdated.

2.6.5 Symbian

Symbian is one of the older OS in the category, with its first release taking place in 1997 as EPOC 32 and discontinued after January 2013. Applications are mainly written in Java, while its native language is Symbian C++. Since many different versions of the OS exist, it is inevitable that slight variations concerning its architecture will also be present (Mokhonoana & Olivier, 2007).

The study of Mokhonoana and Olivier, 2007 discussed the development of an on-phone forensic logical acquisition tool for the Symbian OS (V. 7), which is founded on the dd technique on portable devices running Linux. At first, they produced an introduction to Symbian OS characteristics and then classified potential acquisition methods. Their plan of attack consisted of manual acquisition, use of forensic tools, logical acquisition, including a connection factor, physical acquisition and data gained from service providers (Mokhonoana & Olivier, 2007).

The research objective of Breeuwsma & Jongh (2007) was the physical acquisition of flash memory from different types of embedded systems, mobile devices included. They firstly introduced the features of physical acquisition techniques (chip-off, JTAG, pseudo-physical) and enumerated the advantages and disadvantages of each.Later, they highlighted the importance of "identifying the sectors of data as used by the high level file system before any sort of file system analysis. Moreover, they developed and used a python script, ListLSN, that facilitated the reconstruction of memory blocks by checking and screening out the logical sector numbers (LSNs).

2.7 Embedded mobile storage

Mobile phones contain two types of memories, NOR and NAND flash which are employed to store information. The NOR flash memory was introduced by Intel in the year 1988 while NAND was introduced by Toshiba in 1989. Byte by byte flash memory can be written, but it holds to be wiped out in blocks before it can be re-written to store other information. Erased block is always carved up into pages in NAND flash memory, for instance 32 or 64 pages per erased block. Memory pages are in multiple of 512 bytes in size, furthermore, every page in flash memory has an area of bytes, frequently referred to as the terminated part or spare area. Table 2.2 showed the spare area sizes for different page sizes. The free area can carry data on the condition of the blockage or the page. For instance, when a block turns bad, it will be marked here (Breeuwsma & Jongh, 2007). Nevertheless, this was archived because data and code are stored in NAND flash devices, NAND are available in 128Mb to 1Tb densities for

Table 2.2: Example of spare area sizes for different page sizes (in bytes) Breeuwsma & Jongh (2007).

Page Size	Spare area size	Total page size	Block size
256	8	264	8448
512	16	528	16896
1026	32	1056	33792
2052	64	2112	135168

Table 2.3: Differences between NAND and NOR ELNEC (2013).

	NAND	NOR
Capacity* 1	~ 32Gbit	~1Gbit
Access method	Sequential	Random
Interface	I/O interface	Full memory interface
Performance	Fast read (serial access cycle) Fast write Fast erase (approx. 2ms/block) *2	Fast read (random access) Slow write Slow erases (approx. 1s/block) * 3
Life Span	100 000 – 1 000 000	10 000 – 100 000
Price	Low	High

packaged products, while NOR Flash devices are primarily used for dependable code storage (boot, application, OS). They are available in densities up to 2 GB. There is a big difference between the memories of NOR and NAND flash. Table 2.3 elaborated on the differences.

A spare area in NAND flash can also contain error correction code (ECC) data. ECC data is applied to find errors in a page. Through ECC data an error of one bit can be evened up, after which the blockage will be labelled bad. Lastly spare area can contain evidence necessary for the physical to logical address mapping. All store locations are guaranteed to be good in NOR Flash and to possess the same stage of endurance, thus a relatively great amount of extra memory cells is constructed on the expire pages, these are given to repair defects in the memory array in order to produce a device that possesses all good memory locations. In order to keep costs down and improve yields, the NAND Flash devices contain randomly located invalid blocks in the array. These blocks must be identified before programming the device to avoid losing data stored in the bad memory cells (Wu *et al.*, 2013). Major differences between NAND and NOR are shown in Table 2.3.

2.7.1 Invalid/bad block

From the time when NAND architecture was thought to serve as a low-cost mass storage medium, the standard specification for the NAND allows the existence of invalid/bad blocks in a certain portion (less than 2% maximum). The block is marked as invalid when bad memory location is found (Chen, 2007). Invalid blocks can be assorted into two groups: inherent invalid blocks and acquired invalid blocks. Inherent invalid blocks arise during the production process at the factory. This includes a failure of intentionally isolated block type and/or cell failures, which occur during electrical test. These blocks are identified in invalid block information at the time of shipment of Flash, for maximum number of inherent invalid blocks. Blocks that are considered to be invalid are marked, usually by writing non FFh value (typically 00h) in byte 517 in the first two pages of an invalid block (ELNEC, 2013). Additionally Figure 2.3 illustrated on how a bad block is marked.

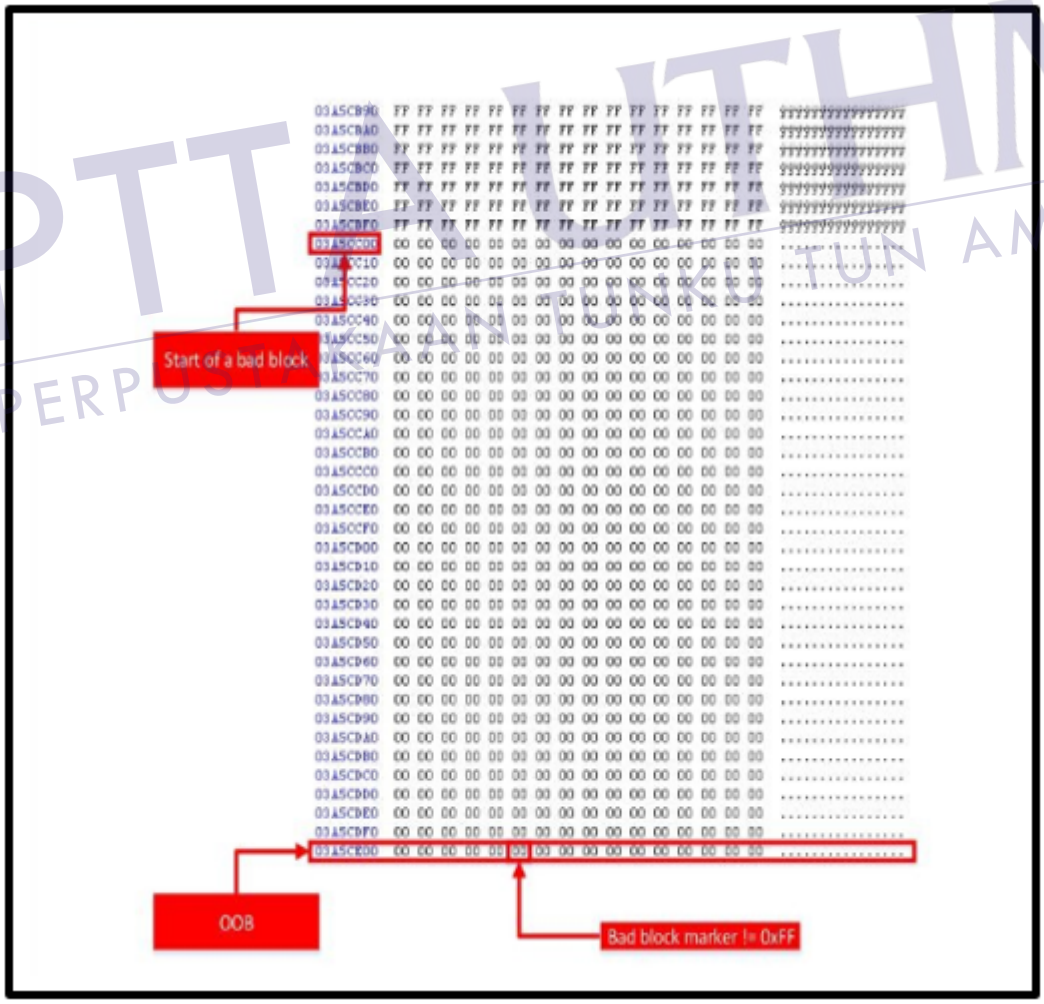

Figure 2.3: How a bad block is marked Wook & Oh (1980).

2.7.2 Bad block management

Multiple bit errors that did not allow read/write or recovery in NAND flash storage is considered as bad block (Wells *et al.*, 2000). This error can lead to several prob lems that can mislead the structure for extraction of dump file from damaged mobile phone, because in that situation there is no Error Correction Code to handle such a case which is usually bound in all new mobile operating system (OS) such as YAFFS2. In addition, ELNEC (2013) mention that to overcome random bit errors, the error detect- ing/correcting algorithm (ECC) must be applied. The author further highlighted reasons for using the ECC algorithm , as stated below. for. For instance, the ECC algorithm is capable for correcting 4 single-bit errors in the frame of 512 bytes which is compulsory for new multi-level cell (MLC) devices (ELNEC, 2013). Still, this is merely applicable to non-damaged mobile phone. In order to overcome such problem on damaged mobile phone, an algorithm is introduced which should be integrated with all MF tools and enable them to further handle bad block issues. The algorithm is illustrated in Figure 3.10 in chapter 3.

2.8 Evidence in mobile phones

Evidence in mobile phones is valuable information that can be used to construct a time-line, compile a list of assistants, or prove intent. Jansen & Ayers (2007) identified the following potential evidence on mobile phones which are as follows: subscriber and equipment identifiers, text messages and multimedia messages, dialed, incoming, and missed call logs, electronic mail, date/time, language, and other settings, phone book information, appointment calendar information, photos, audio and video recordings, instant messaging and web browsing activities, electronic documents, location information . In the list of potential evidence above, two from the category will be the main focus in this research which are images that comprise JPEG, Exif, JFIF and multimedia files such as 3gp and MP4 videos, which are the most common file types found in mobile phones. However the images and videos selected in this research are common in terms of file structure, for instance both image types contain the same header and footer known as “FFD8” and “FFD9” while the 3gp and MP4 file has the same container known as ATOM which will be discussed in detail.

REFERENCES

Abdullah, N.A., Ibrahim, R. & Mohamad, K.M. (2013). Carving Thumbnail/s and Embedded JPEG Files Using Image Pattern Matching. *Journal of Software Engineering and Applications*, 6(3B), pp. 62–66.

Abdullah, N.A., Ibrahim, R. & Mohamad, K.M. (2014). *An IMPROVE file carver of intertwined jpeg images using X-mykarve*. Ph.D. thesis, University Tun Hussein Onn Malaysia.

Ahmed, R., Dharaskar, R.V. & Thakare, V.M. (2013). Digital evidence extraction and documentation from mobile devices. 2(1), pp. 1019–1024.

Akkaladevi, S., Keesara, H., Christi, C. & Luo, X. (2011). Efficient forensic tools for handheld devices :, pp. 349–359.

Al Barghouthy, N., Marrington, A. & Baggili, I. (2013). The forensic investigation of android private browsing sessions using orweb. In: *Computer Science and Information Technology (CSIT), 2013 5th International Conference on IEEE*, pp. 33–37.

Al-zarouni, M. (2006). Mobile Handset Forensic Evidence : a challenge for Law Enforcement.

Ayers, R., Brothers, S. & Jansen, W. (2014). Guidelines on mobile device forensics.

Azadegan, S., Yu, W., Liu, H., Sistani, M. & Acharya, S. (2012). Novel Anti-forensics Approaches for Smart Phones. *2012 45th Hawaii International Conference on System Sciences*, pp. 5424–5431.

Barmpatsalou, K., Damopoulos, D., Kambourakis, G. & Katos, V. (2013a). A critical review of 7 years of Mobile Device Forensics. *Digital Investigation*, 10(4), pp. 323–349.

Barmpatsalou, K., Damopoulos, D., Kambourakis, G. & Katos, V. (2013b). A critical review of 7 years of Mobile Device Forensics. *Digital Investigation*, 10(4), pp. 323–349.

Bashir, M. & Khan, M. (2013). Triage in Live Digital Forensic Analysis. *The International Journal of Forensic Computer Science*, 8(1), pp. 35–44.

Becker, A., Mladenow, A., Kryvinska, N. & Strauss, C. (2012). Aggregated survey of sustainable business models for agile mobile service delivery platforms. *Journal of Service Science Research Springer*, 4(1), pp. 97–121.

Beek, B.C., Consultant, P.S. & Services, P. (2011). Introduction to File carving. White paper. McAfee.

Bekhet, S., Ahmed, A., Hunter, A. *et al.* (2013). Video matching using dc-image and local features. *Lecture Notes in Engineering and Computer Science Newswood Limited/International Association of Engineers*, 3, pp. 2209–2214.

Bert, R. (2010). *Tecniche di Triage applicate alla Digital Forensics*. Ph.D. thesis, PhD Thesis. Universit degli Studi di Roma Tor Vergata.

Blokhuis, J. & Puppe, A. (2010). *Research Project 2 DFRWS Challenge 2010-Mobile forensics*. Citeseer.

Breeuwsma, M.F. (2006). Forensic imaging of embedded systems using JTAG (boundary-scan). *Digital Investigation*, 3(1), pp. 32–42.

Breeuwsma, M. & Jongh, M.D. (2007). Forensic data recovery from flash memory. *Small Scale Digital . . .* , 1(1), pp. 1–17.

Brinson, A., Robinson, A. & Rogers, M. (2006). A cyber forensics ontology: Creating a new approach to studying cyber forensics. *digital investigation Elsevier*, 3, pp. 37–43.

Cantrell, G., George, S. & Dampier, D.A. (2012). Implementing the automated phases of the partially-automated digital triage process model. 7(4), pp. 99–116.

Casey, E. (2004). *Digital evidence and computer crime-forensics science, computers and the internet 2. nd edition*. Elsevier academic press.

Casey, E., Cheval, A., Lee, J.Y., Oxley, D. & Song, Y.J. (2011). Forensic acquisition and analysis of palm webOS on mobile devices. *Digital Investigation*, 8(1), pp. 37–47.

Casey, E. & Turnbull, B. (2011). Digital evidence on mobile devices. *Eoghan Casey, Digital Evidence and Computer Crime. Third Edition. Forensic Science, Computers, and the Internet, Academic Pres*.

CCITT, S. (1992). Working party. *Experts Group for ATM Video Coding, Working Document AVC-205*.

Chang, C.P., Chen, C.T., Lu, T.H., Lin, I.L., Huang, P. & Lu, H.S. (2013). Study on constructing forensic procedure of digital evidence on smart handheld device. *2013 International Conference on System Science and Engineering (ICSSE)*, pp. 223–228.

Chen, F., Koufaty, D.A. & Zhang, X. (2009). Understanding intrinsic characteristics and system implications of flash memory based solid state drives. In: *ACM SIGMETRICS Performance Evaluation Review*, volume 37, pp. 181–192.

Chen, S. (2007). What types of ecc should be used on flash memory. *Application Note for SPANSION*.

Christi, C., Mallepally, R., Members, C., Chairperson, C. & Member, C. (2011). Implementation of applications to improve iphone forensic analysis and integrity of evidence.

Curran, K., Robinson, A., Peacocke, S. & Cassidy, S. (2010). Mobile Phone Forensic Analysis. 2(2).

Dubey, S. (2013). Mobile Forensics and Damage Recovery of Forensics Evidence Images. 1(1), pp. 10–15.

ELNEC (2013). NAND Flash Memories Application Note NAND Flash Memories and Programming NAND Flash Memories Using ELNEC Device Programmers, pp. 1–44.

ePrint Archive, I.C. (2011). Higher-order differential attack on reduced sha-256. *Lamberger, Mario and Mendel, Florian*, 2011, p. 37.

Faheem, M., Le-Khac, N.A. & Kechadi, T. (2014). Smartphone forensic analysis: A case study for obtaining root access of an android samsung s3 device and analyse the image without an expensive commercial tool. *Journal of Information Security, Scientific Research Publishing*, 2014.

Fairbanks, K., Atreya, K. & Owen, H. (2009). Blackberry ipd parsing for open source forensics. In: *Southeastcon, 2009. SOUTHEASTCON'09. IEEE*, pp. 195–199.

Fp-sec crisalis, C.N. (2013). D7.1 Forensics. 1.

Garfinkel, S.L. (2013). Digital media triage with bulk data analysis and bulk-extractor. *Computers and Security*, 32, pp. 56–72.

Gligoroski, D., Markovski, S. & Kocarev, L. (2009). Edon-r, an infinite family of cryptographic hash functions. *IJ Network Security*, 8(3), pp. 293–300.

Gottschalk, P. (2010). *Policing Cyber Crime*. Bookboon.

Grispos, G., Storer, T. & Bradley, W. (2012). A Comparison of Forensic Evidence Recovery Techniques for a Windows Mobile Smart Phone. 8(November), pp. 23–36.

He, M.T. & Tehranipoor, M. (2014). A comprehensive mechanism for accessing embedded sensors in modern socs. In: *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2014 IEEE International Symposium on*, pp. 240–245.

Hong, I., Yu, H., Lee, S. & Lee, K. (2013). A new triage model conforming to the needs of selective search and seizure of electronic evidence. *Digital Investigation*, 10(2), pp. 175–192.

Hoog, A. (2011). Android forensic techniques. *Android Forensics. Boston: Syngress*, pp. 195–284.

Hoog, A. & Gaffaney, K. (2009). iphone forensics. *Via Forensics White paper*.

Husain, M.I., Baggili, I. & Sridhar, R. (2011). A simple cost-effective framework for iphone forensic analysis. In: *Digital Forensics and Cyber Crime, Springer*, pp. 27–37.

Jansen, W. & Ayers, R. (2007). Guidelines on cell phone forensics. *NIST Special Publication*, 800, p. 101.

K mmerling, O. & Kuhn, M.G. (1999). Design principles for tamper-resistant smartcard processors. In: *USENIX workshop on Smartcard Technology*, pp. 9–20.

Kaart, M., Klaver, C. & van Baar, R. (2013). Forensic access to Windows Mobile pim.vol and other Embedded Database (EDB) volumes. *Digital Investigation*, 9(3-4), pp. 170–192.

Kalva, H., Parikh, A. & Srinivasan, A. (2013). Accelerating Video Carving from Unallocated Space. *Media Watermarking Security and Forensics*, 8665, pp. 1–4.

Karabiyik, U. (2015). *Building an Intelligent Assistant for Digital Forensics*. Ph.D. thesis, Florida State University.

Kim, D., Park, J., Lee, K.g. & Lee, S. (2012). Forensic analysis of android phone using ext4 file system journal log. In: *Future Information Technology, Application, and Service. Springer*, pp. 435–446.

Kizza, J.M. (2010). *Computer crime investigations–computer forensics*. Springer.

Klaver, C. (2010). Windows mobile advanced forensics. *digital investigation, Elsevier*, 6(3), pp. 147–167.

Kloet, S. *et al.* (2007). Measuring and improving the quality of file carving methods. *Almere, Niederlande: Eindhoven University of Technology*, pp. 4–79.

Kornblum, J.D. (2008). Using jpeg quantization tables to identify imagery processed by software. *Digital Investigation Elsevier*, 5, pp. S21–S25.

Kumar, S.S., Thomas, B. & Thomas, K. (2012). An agent based tool for windows mobile forensics. In: *Digital Forensics and Cyber Crime, Springer*, pp. 77–88.

Lai, Y., Yang, C., Lin, C. & Ahn, T. (2011). Design and implementation of mobile forensic tool for android smart phone through cloud computing. In: *Convergence and Hybrid Information Technology Springer*, pp. 196–203.

Law, F.Y., Chan, P.P., Yiu, S.M., Chow, K.P., Kwan, M.Y., Tse, H.K. & Lai, P.K. (2011). Protecting digital data privacy in computer forensic examination. In: *Systematic Approaches to Digital Forensic Engineering (SADFE), 2011 IEEE Sixth International Workshop on*, pp. 1–6.

Lessard, J., Kessler, G. & Kessler, G.C. (2011). Android Forensics : Simplifying Cell Phone Examinations . Android Forensics : Simplifying Cell Phone Examinations. 4(2010), pp. 1–12.

Luk, R.W., Damper, R. *et al.* (1992). Inference of letter-phoneme correspondences by delimiting and dynamic time warping techniques. In: *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 2, pp. 61–64.

Luo, Z., Zheng, Q., Hei, X. & Giacaman, N. (2013). Parallel programming based on microsoft. net tpl. In: *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering*, Atlantis Press.

Marturana, F. & Tacconi, S. (2013). A Machine Learning-based Triage methodology for automated categorization of digital media. *Digital Investigation*, 10(2), pp. 193–204.

Meyers, M. & Rogers, M. (2004). Computer forensics: the need for standardization and certification. *International Journal of Digital Evidence*, 3(2), pp. 1–11.

Mohamad, K.M. & Deris, M.M. (2009). Visualization of jpeg metadata. In: *Visual Informatics Bridging Research and Practice Springer*, pp. 543–550.

Mohamad, K.M., Herawan, T. & Deris, M.M. (2011). Detecting JFIF header using FORHEADER. *International Journal of Security and its Applications*, 5(4), pp. 23–36.

Mokhonoana, P.M. & Olivier, M.S. (2007). Acquisition of a symbian smart phones content with an on phone forensic tool. In: *Proceedings of the Southern African Telecommunication Networks and Applications Conference*, volume 8.

Mrdovic, S., Huseinovic, A. & Zajko, E. (2009). Combining static and live digital forensic analysis in virtual environment. In: *Information, Communication and Automation Technologies, 2009. ICAT 2009. XXII International Symposium on IEEE*, pp. 1–6.

Muhammad, A. & Ashraf, N. (2012). Forensic Multimedia File Carving.

Naresh, Y.R.P.S.G.K.K.K.T. (2013). Enhancing the integrity of short message service (sms) in new generation mobile devices. *International Journal of Computer Science Issues*, 10(6), pp. 282–288.

Olivier, M.S. (2009). On metadata context in database forensics. *Digital Investigation Elsevier*, 5(3), pp. 115–123.

Owen, P. & Thomas, P. (2011). An analysis of digital forensic examinations: Mobile devices versus hard disk drives utilising ACPO & NIST guidelines. *Digital Investigation*, 8(2), pp. 135–140.

Pal, A. & Memon, N. (2009). The Evolution of File Carving [. (March), pp. 59–71.

Park, J., Chung, H. & Lee, S. (2012). Forensic analysis techniques for fragmented flash memory pages in smartphones. *Digital Investigation*, 9(2), pp. 109–118.

Pearson, S. & Watson, R. (2010). *Digital triage forensics: processing the digital crime scene*. Syngress.

Psannis, K.E. & Ishibashi, Y. (2005). Mpeg-4 interactive video streaming over wireless networks. *WSEAS Transactions on Information Science and Applications*, 2(8), pp. 1131–1137.

Quick, D. & Alzaabi, M. (2011a). Forensic analysis of the android file system YAFFS2. *Proceedings of the 9th Australian Digital Forensics Conference*, (December), pp. 100–109.

Quick, D. & Alzaabi, M. (2011b). Forensic analysis of the android file system YAFFS2. (December).

Richardson, I.E. (2011). *The H. 264 advanced video compression standard*. John Wiley & Sons.

Rogers, M.K., Mislan, R., Goldman, J., Wedge, T. & Debrota, S. (2006). Computer Forensics Field Triage Process Model. *Conference on Digital Forensics, Security and Law*, 1(2), pp. 27–40.

Simao, A.M.D.L. & Sicoli, F.C. (2011). Acquisition of digital evidence in android smartphones. (December).

Spreitzenbarth, M. & Freiling, F.C. (2012). Forensic Recovery of Scrambled Telephones, pp. 1–19.

Supriya Kulkarni, P. & Jisha, P. (2013). Study of bad block management and wear leveling in nand flash memories. *International Journal of Research in Engineering and Technology (IJRET)*, 2(10).

Sylve, J., Case, A., Marziale, L. & Richard, G.G. (2012). Acquisition and analysis of volatile memory from android devices. *Digital Investigation*, 8(3-4), pp. 175–184.

Thakur, N.S. (2013). Forensic Analysis of WhatsApp on Android Smartphones.

Thing, V.L., Ng, K.Y. & Chang, E.C. (2010a). Live memory forensics of mobile phones. *digital investigation Elsevier*, 7, pp. S74–S82.

Thing, V.L., Ng, K.Y. & Chang, E.C. (2010b). Live memory forensics of mobile phones. *Digital Investigation*, 7, pp. S74–S82.

Vasa, T.S. (2013). Mobile Phone : Identifying Configuration Signatures of Local Devices Absent from XRY.

Vidas, T., Zhang, C. & Christin, N. (2011). Toward a general collection methodology for Android devices. 8.

Viterbi, A.J. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions on*, 13(2), pp. 260–269.

Walls, R.J., Learned-miller, E. & Levine, B.N. (2011). Forensic Triage for Mobile Phones with DEC0DE.

Walls, R.J. & Levine, B.N. (2014). Efficient Smart Phone Forensics Based on Relevance Feedback.

Wells, S.E., Magnusson, E.J. & Hasbun, R.N. (2000). *Method of managing defects in flash disk memories*. Google Patents, uS Patent 6,014,755.

Wook, J. & Oh, M. (1980). Reverse Engineering Flash Memory for Fun and Benefit.

Wu, B., Xu, M., Zhang, H., Xu, J., Ren, Y. & Zheng, N. (2013). A Recovery Approach for SQLite History Recorders from YAFFS2 Recovery Elements of SQLite Deleted Records, pp. 295–299.

Yang, C.H. & Yen, P.H. (2010). Fast deployment of computer forensics with usbs. In: *Broadband, Wireless Computing, Communication and Applications (BWCCA), 2010 International Conference on IEEE*, pp. 413–416.

Yates, I. *et al.* (2010). Practical investigations of digital forensics tools for mobile devices. In: *2010 Information Security Curriculum Development Conference*.

Zdziarski, J. (2008). *iPhone forensics: recovering evidence, personal data, and corporate assets*. O'Reilly Media, Inc.

Zimmermann, C., Spreitzenbarth, M., Schmitt, S. & Freiling, F.C. (2012). Forensic analysis of yaffs2. In: *Sicherheit*, pp. 59–69.